Distributive Aronszajn trees

The 14th International Workshop on Set Theory
CIRM, Luminy, Marseille
10-October-2017

Assaf Rinot
Bar-Ilan University
**Conventions**

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal.
Conventions

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal. Often times, $\kappa = \lambda^+$. 
Conventions

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal. Often times, $\kappa = \lambda^+$. 

$H_\kappa$ denotes the collection of all sets of hereditary cardinality $< \kappa$. 

$\Kappa(\kappa)$ denotes the collection of all $x \in \mathcal{P}(\kappa)$ such that $x$ is a club subset of $\text{sup}(x)$. 

Every set of ordinals $C$, splits into two:

- $\text{acc}(C) := \{\alpha \in C | \text{sup}(C \cap \alpha) = \alpha > 0\}$;
- $\text{nacc}(C) := C \setminus \text{acc}(C)$.

When we write "there is a limit $\alpha < \kappa$", we mean "$\exists \alpha \in \text{acc}(\kappa)$".
Conventions

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal. Often times, $\kappa = \lambda^+$. 

$H_\kappa$ denotes the collection of all sets of hereditary cardinality $< \kappa$. 

$\mathcal{K}(\kappa)$ denotes the collection of all $x \in \mathcal{P}(\kappa)$ such that $x$ is a club subset of $\text{sup}(x)$. 
Conventions

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal. Often times, $\kappa = \lambda^+$. 

$H_\kappa$ denotes the collection of all sets of hereditary cardinality $< \kappa$.

$\mathcal{K}(\kappa)$ denotes the collection of all $x \in \mathcal{P}(\kappa)$ such that $x$ is a club subset of $\text{sup}(x)$.

Every set of ordinals $C$, splits into two:

- $\text{acc}(C) := \{ \alpha \in C \mid \text{sup}(C \cap \alpha) = \alpha > 0 \}$;
- $\text{nacc}(C) := C \setminus \text{acc}(C)$. 
Conventions

Throughout, $\kappa$ denotes a regular uncountable cardinal, and $\lambda$ denotes an uncountable cardinal. Often times, $\kappa = \lambda^+$.

$H_\kappa$ denotes the collection of all sets of hereditary cardinality $< \kappa$.

$\mathcal{K}(\kappa)$ denotes the collection of all $x \in \mathcal{P}(\kappa)$ such that $x$ is a club subset of $\text{sup}(x)$.

Every set of ordinals $C$, splits into two:

- $\text{acc}(C) := \{\alpha \in C \mid \text{sup}(C \cap \alpha) = \alpha > 0\}$;
- $\text{nacc}(C) := C \setminus \text{acc}(C)$.

When we write “there is a limit $\alpha < \kappa$”, we mean “$\exists \alpha \in \text{acc}(\kappa)$”.
Definition
In this talk, a $\kappa$-tree is a nonempty subset $T \subseteq {}^{<\kappa}H_\kappa$, satisfying:

1. for all $\alpha < \kappa$, the set $T_\alpha := T \cap {}^\alpha H_\kappa$ has size $< \kappa$;
2. for all $\alpha < \kappa$ and $t \in T$, there is $s \in T_\alpha$ such that $t \cup s \in T$. 

To each $T$, we associate the notion of forcing $P(T) := (T, \supseteq)$. 
\( \kappa \)-trees

**Definition**

In this talk, a \( \kappa \)-tree is a nonempty subset \( T \subseteq {<\kappa}^\kappa \), satisfying:

1. for all \( \alpha < \kappa \), the set \( T_\alpha := T \cap {^\alpha}^\kappa \) has size \( < \kappa \);
**Definition**

In this talk, a $\kappa$-tree is a nonempty subset $T \subseteq {}^{<\kappa}H_\kappa$, satisfying:

1. for all $\alpha < \kappa$, the set $T_\alpha := T \cap {}^\alpha H_\kappa$ has size $< \kappa$;
2. for all $\alpha < \kappa$ and $t \in T$, there is $s \in T_\alpha$ such that $t \cup s \in T$. 

To each $T$, we associate the notion of forcing $P(T) := (T, \supseteq)$. 

kappa-trees
\(\kappa\)-trees

**Definition**

In this talk, a \(\kappa\)-tree is a nonempty subset \(T \subseteq {^{<\kappa}} H_\kappa\), satisfying:

1. for all \(\alpha < \kappa\), the set \(T_\alpha := T \cap {^\alpha} H_\kappa\) has size \(< \kappa\);  
2. for all \(\alpha < \kappa\) and \(t \in T\), there is \(s \in T_\alpha\) such that \(t \cup s \in T\).

To each \(T\), we associate the notion of forcing \(\mathbb{P}(T) := (T, \supseteq)\).
\( \kappa \)-trees

**Definition**
In this talk, a \( \kappa \)-tree is a nonempty subset \( T \subseteq <\kappa H_\kappa \), satisfying:

1. for all \( \alpha < \kappa \), the set \( T_\alpha := T \cap \alpha H_\kappa \) has size \( < \kappa \);
2. for all \( \alpha < \kappa \) and \( t \in T \), there is \( s \in T_\alpha \) such that \( t \cup s \in T \).

To each \( T \), we associate the notion of forcing \( \mathbb{P}(T) := (T, \supseteq) \).

**Note**
If \( T \) is a \( \kappa \)-tree, then \( \mathbb{P}(T) \) adds a cofinal branch through \( T \). i.e., a sequence \( b : \kappa \rightarrow H_\kappa \) such that \( b \upharpoonright \alpha \in T \) for all \( \alpha < \kappa \).
\(\kappa\)-trees

**Definition**

In this talk, a \(\kappa\)-tree is a nonempty subset \(T \subseteq <\kappa H_\kappa\), satisfying:

1. for all \(\alpha < \kappa\), the set \(T_\alpha := T \cap ^\alpha H_\kappa\) has size \(<\kappa\);
2. for all \(\alpha < \kappa\) and \(t \in T\), there is \(s \in T_\alpha\) such that \(t \cup s \in T\).

To each \(T\), we associate the notion of forcing \(\mathbb{P}(T) := (T, \supseteq)\).

**Definition**

A \(\kappa\)-tree \(T\) is **Aronszajn** iff it has no cofinal branches.

**Note**

If \(T\) is a \(\kappa\)-tree, then \(\mathbb{P}(T)\) adds a cofinal branch through \(T\). i.e., a sequence \(b : \kappa \rightarrow H_\kappa\) such that \(b \upharpoonright \alpha \in T\) for all \(\alpha < \kappa\).
Definition
In this talk, a \( \kappa \)-tree is a nonempty subset \( T \subseteq {}^{<\kappa}H_\kappa \), satisfying:

1. for all \( \alpha < \kappa \), the set \( T_\alpha := T \cap {}^{\alpha}H_\kappa \) has size \( < \kappa \);
2. for all \( \alpha < \kappa \) and \( t \in T \), there is \( s \in T_\alpha \) such that \( t \cup s \in T \).

To each \( T \), we associate the notion of forcing \( \mathbb{P}(T) := (T, \supseteq) \).

Definition
A \( \kappa \)-tree \( T \) is **Aronszajn** iff it has no cofinal branches.

Definition
A \( \kappa \)-tree \( T \) is **Souslin** iff it is Aronszajn and \( \mathbb{P}(T) \) has the \( \kappa \)-cc.
λ⁺-Souslin trees

Jensen proved that, in $L$, for all (regular uncountable) $\kappa$ that is not weakly compact, there is a $\kappa$-Souslin tree.
Jensen proved that, in $L$, for all (regular uncountable) $\kappa$ that is not weakly compact, there is a $\kappa$-Souslin tree. His proof shows:

**Theorem (Jensen, 1972)**

*For all uncountable $\lambda$, $\text{GCH} + \square_\lambda$ yields a $\lambda^+$-Souslin tree.*
Jensen proved that, in $L$, for all (regular uncountable) $\kappa$ that is not weakly compact, there is a $\kappa$-Souslin tree. His proof shows:

**Theorem (Jensen, 1972)**

*For all uncountable $\lambda$, GCH $+ \square_\lambda$ yields a $\lambda^+$-Souslin tree.*

This was recently improved:

**Theorem (Rinot, 2017)**

*For all uncountable $\lambda$, GCH $+ \square(\lambda^+)$ yields a $\lambda^+$-Souslin tree.*
$\lambda^+$-Souslin trees

A tree $T$ is coherent iff $\{ \alpha \in \text{dom}(s) \cap \text{dom}(t) \mid s(\alpha) \neq t(\alpha) \}$ is finite for all $s, t \in T$.

Theorem (Jensen, 1972)

For all uncountable $\lambda$, $\text{GCH} + \Box_{\lambda}$ yields a $\lambda^+$-Souslin tree.

This was recently improved:

Theorem (Rinot, 2017)

For all uncountable $\lambda$, $\text{GCH} + \Box(\lambda^+)$ yields a $\lambda^+$-Souslin tree.
$\lambda^+$-Souslin trees

A tree $T$ is coherent iff $\{ \alpha \in \text{dom}(s) \cap \text{dom}(t) \mid s(\alpha) \neq t(\alpha) \}$ is finite for all $s, t \in T$.

**Theorem (Veličković, 1986)**

*For all uncountable $\lambda$, $\diamondsuit_\lambda$ yields a coherent $\lambda^+$-Souslin tree.*

This was recently improved:

**Theorem (Rinot, 2017)**

*For all uncountable $\lambda$, GCH + $\square(\lambda^+)$ yields a $\lambda^+$-Souslin tree.*
A tree $T$ is **coherent** iff $\{\alpha \in \text{dom}(s) \cap \text{dom}(t) \mid s(\alpha) \neq t(\alpha)\}$ is finite for all $s, t \in T$.

**Theorem (Veličković, 1986)**

*For all uncountable $\lambda$, $\Box_\lambda$ yields a coherent $\lambda^+$-Souslin tree.*

This was recently improved:

**Theorem (Rinot, 2017)**

*For all uncountable $\lambda$, GCH + $\square(\lambda^+)$ yields a $\lambda^+$-Souslin tree.*

Even more recently:

**Theorem (Brodsky-Rinot, 201∞)**

*For all singular $\lambda$, GCH + $\square(\lambda^+)$ yields a coherent $\lambda^+$-Souslin tree.*
λ⁺-Souslin trees

In this talk, I would like to discuss the techniques that go into the proofs, and to report on progress made on a related problem.

This was recently improved:

Theorem (Rinot, 2017)

For all uncountable λ, GCH + □(λ⁺) yields a λ⁺-Souslin tree.

Even more recently:

Theorem (Brodsky-Rinot, 201∞)

For all singular λ, GCH + □(λ⁺) yields a coherent λ⁺-Souslin tree.
A related problem

Definition
A $\kappa$-tree $T$ is **collapsing** iff $\mathbb{P}(T)$ collapses cardinals.
A related problem

Definition
A $\kappa$-tree $T$ is **collapsing** iff $\mathbb{P}(T)$ collapses cardinals.

Collapsing Tree Property
CTP($\kappa$) asserts that the two hold:
1. there exists a $\kappa$-Aronszajn tree;
2. every $\kappa$-Aronszajn tree is collapsing.
A related problem

Definition
A $\kappa$-tree $T$ is **collapsing** iff $\mathbb{P}(T)$ collapses cardinals.

Collapsing Tree Property
CTP($\kappa$) asserts that the two hold:
1. there exists a $\kappa$-Aronszajn tree;
2. every $\kappa$-Aronszajn tree is collapsing.

Theorem (Jensen, 1970’s)
GCH *is consistent with* CTP($\aleph_1$).
A related problem

Definition
A $\kappa$-tree $T$ is **collapsing** iff $\mathcal{P}(T)$ collapses cardinals.

Collapsing Tree Property
CTP($\kappa$) asserts that the two hold:
1. there exists a $\kappa$-Aronszajn tree;
2. every $\kappa$-Aronszajn tree is collapsing.

Theorem (Jensen, 1970’s)
GCH is consistent with CTP($\aleph_1$).

Theorem (Laver-Shelah, 1981)
Assuming a weakly compact, CH is consistent with CTP($\aleph_2$).
A related problem

**Conjecture**
For every uncountable cardinal $\lambda$, $\text{GCH} \iff \neg \text{CTP}(\lambda^+)$. 

**Collapsing Tree Property**
$\text{CTP}(\kappa)$ asserts that the two hold:
1. there exists a $\kappa$-Aronszajn tree;
2. every $\kappa$-Aronszajn tree is collapsing.

**Theorem (Jensen, 1970’s)**
$\text{GCH}$ is consistent with $\text{CTP}((\aleph_1))$.

**Theorem (Laver-Shelah, 1981)**
Assuming a weakly compact, $\text{CH}$ is consistent with $\text{CTP}((\aleph_2))$. 
A related problem

Conjecture
For every uncountable cardinal $\lambda$, GCH $\implies \neg CTP(\lambda^+)$.  

Collapsing Tree Property
CTP($\kappa$) asserts that the two hold:
1. there exists a $\kappa$-Aronszajn tree;
2. every $\kappa$-Aronszajn tree is collapsing.

It is now inevitable to discuss square principles...
Square principles
Square principles

Definition (Jensen, 1972)

□\(\lambda\): exists a sequence \(\langle C_\alpha \mid \alpha < \lambda^+ \rangle\) such that for every limit \(\alpha\):

1. \(C_\alpha\) is a club in \(\alpha\) of order-type \(\leq \lambda\);
2. for all \(\bar{\alpha} \in \text{acc}(C_\alpha)\), \(C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}}\).
Square principles

Definition (Jensen, 1972)
□_\lambda: exists a sequence \langle C_\alpha \mid \alpha < \lambda^+ \rangle such that for every limit \alpha:
1. \( C_\alpha \) is a club in \( \alpha \) of order-type \( \leq \lambda \);
2. for all \( \bar{\alpha} \in \text{acc}(C_\alpha) \), \( C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}} \).

We generalize the preceding from a cardinal \( \lambda \) to an ordinal \( \xi \):

Definition
□_\xi: exists a sequence \langle C_\alpha \mid \alpha < |\xi|^+ \rangle such that for every limit \( \alpha \):
1. \( C_\alpha \) is a club in \( \alpha \) of order-type \( \leq \xi \);
2. for all \( \bar{\alpha} \in \text{acc}(C_\alpha) \), \( C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}} \).
Wait a minute!
But for all $\xi \in [\lambda, \lambda^+]$, $\square_\xi$ is equivalent to $\square_\lambda$.

We generalize the preceding from a cardinal $\lambda$ to an ordinal $\xi$:

**Definition**

$\square_\xi$: exists a sequence $\langle C_\alpha \mid \alpha < |\xi|^+ \rangle$ such that for every limit $\alpha$:

1. $C_\alpha$ is a club in $\alpha$ of order-type $\leq \xi$;
2. for all $\bar{\alpha} \in \text{acc}(C_\alpha)$, $C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}}$. 
Square principles

Wait a minute!
But for all $\xi \in [\lambda, \lambda^+]$, $\Box_\xi$ is equivalent to $\Box_\lambda$.
It’s true, but we nevertheless claim that $\Box_{\lambda^2}$ is superior over $\Box_\lambda$.

We generalize the preceding from a cardinal $\lambda$ to an ordinal $\xi$:

Definition
$\Box_\xi$: exists a sequence $\langle C_\alpha \mid \alpha < |\xi|^+ \rangle$ such that for every limit $\alpha$:

1. $C_\alpha$ is a club in $\alpha$ of order-type $\leq \xi$;
2. for all $\bar{\alpha} \in \text{acc}(C_\alpha)$, $C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}}$. 
Wait a minute!

But for all \( \xi \in [\lambda, \lambda^+] \), \( \Box_\xi \) is equivalent to \( \Box_\lambda \).

It’s true, but we nevertheless claim that \( \Box_\lambda^2 \) is superior over \( \Box_\lambda \).

Why? because the former allows \( \{ \alpha \in E_\theta^{\lambda^+} \mid |C_\alpha| = |\alpha| \} \) to be stationary for any choice of a regular cardinal \( \theta \leq \lambda \).

Definition

\( \Box_\xi \): exists a sequence \( \langle C_\alpha \mid \alpha < |\xi|^+ \rangle \) such that for every limit \( \alpha \):

1. \( C_\alpha \) is a club in \( \alpha \) of order-type \( \leq \xi \);
2. for all \( \bar{\alpha} \in \text{acc}(C_\alpha) \), \( C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}} \).
Square principles

Wait a minute!
But for all $\xi \in [\lambda, \lambda^+)$, $\Box_\xi$ is equivalent to $\Box_\lambda$.
It’s true, but we nevertheless claim that $\Box^2_\lambda$ is superior over $\Box_\lambda$.
Why? because the former allows $\{ \alpha \in E^\lambda_\theta \mid |C_\alpha| = |\alpha| \}$ to be stationary for any choice of a regular cardinal $\theta \leq \lambda$.

Definition
$\Box_\xi(\kappa)$: exists a sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ such that for every limit $\alpha$:
1. $C_\alpha$ is a club in $\alpha$ of order-type $\leq \xi$;
2. for all $\bar{\alpha} \in \text{acc}(C_\alpha)$, $C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}}$. 
Square principles

Wait a minute!
But for all $\xi \in [\lambda, \lambda^+]$, $\square_\xi$ is equivalent to $\square_\lambda$.
It’s true, but we nevertheless claim that $\square_\lambda^2$ is superior over $\square_\lambda$.
Why? because the former allows $\{\alpha \in E_{\theta}^{\lambda^+} \mid |C_\alpha| = |\alpha|\}$ to be stationary for any choice of a regular cardinal $\theta \leq \lambda$.

Definition
$\square_\xi(\kappa)$: exists a sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ such that for every limit $\alpha$:
1. $C_\alpha$ is a club in $\alpha$ of order-type $\leq \xi$;
2. for all $\bar{\alpha} \in \text{acc}(C_\alpha)$, $C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}}$;
3. for every club $D \subseteq \kappa$, there is $\bar{\alpha} \in \text{acc}(D)$ with $D \cap \bar{\alpha} \neq C_{\bar{\alpha}}$. 
Square principles

Definition

\( \square_{\xi}(\kappa, < \mu) \): exists a sequence \( \langle C_\alpha \mid \alpha < \kappa \rangle \) such that for limit \( \alpha \):

1. \( C_\alpha \) is a club in \( \alpha \) of order-type \( \leq \xi \);
2. \( C_\alpha := \{ C_\beta \cap \alpha \mid \beta < \kappa, \sup(C_\beta \cap \alpha) = \alpha \} \) has size \( < \mu \);
3. for every club \( D \subseteq \kappa \), there is \( \bar{\alpha} \in \text{acc}(D) \) with \( D \cap \bar{\alpha} \notin C_{\bar{\alpha}} \).

Definition

\( \square_{\xi}(\kappa) \): exists a sequence \( \langle C_\alpha \mid \alpha < \kappa \rangle \) such that for every limit \( \alpha \):

1. \( C_\alpha \) is a club in \( \alpha \) of order-type \( \leq \xi \);
2. for all \( \bar{\alpha} \in \text{acc}(C_\alpha) \), \( C_\alpha \cap \bar{\alpha} = C_{\bar{\alpha}} \);
3. for every club \( D \subseteq \kappa \), there is \( \bar{\alpha} \in \text{acc}(D) \) with \( D \cap \bar{\alpha} \neq C_{\bar{\alpha}} \).
Square principles

Definition
□_ξ(κ, < μ): exists a sequence ⟨C_α | α < κ⟩ such that for limit α:
1. C_α is a club in α of order-type ≤ ξ;
2. C_α := \{C_β ∩ α | β < κ, sup(C_β ∩ α) = α\} has size < μ;
3. for every club D ⊆ κ, there is ¯α ∈ acc(D) with D ∩ ¯α /∈ C_¯α.

Square principles and Aronszajn trees are closely related:
Theorem (Jensen, 1972)
□_λ(λ^+, < λ^+) holds iff there exists a special λ^+-Aronszajn tree.
Square principles

Definition

□_ξ(κ, < μ): exists a sequence ⟨C_α | α < κ⟩ such that for limit α:
1. C_α is a club in α of order-type ≤ ξ;
2. C_α := \{ C_β ∩ α | β < κ, sup(C_β ∩ α) = α \} has size < μ;
3. for every club D ⊆ κ, there is ¯α ∈ acc(D) with D ∩ ¯α ∉ C_¯α.

Square principles and Aronszajn trees are closely related:

Theorem (Jensen, 1972)

□_λ(λ^+, < λ^+) holds iff there exists a special λ^+-Aronszajn tree.

Theorem (Todorcevic, 1987)

□_κ(κ, < κ) holds iff there exists a κ-Aronszajn tree.
Square principles

Recall our conjecture
For every uncountable cardinal $\lambda$, $\text{GCH} \implies \neg \text{CTP}(\lambda^+)$. 

Theorem (Jensen, 1972)
$\square_\lambda(\lambda^+, < \lambda^+) \text{ holds iff there exists a special } \lambda^+\text{-Aronszajn tree.}$

Theorem (Todorcevic, 1987)
$\square_\kappa(\kappa, < \kappa) \text{ holds iff there exists a } \kappa\text{-Aronszajn tree.}$
Square principles

Recall our conjecture
For every uncountable cardinal $\lambda$, $\text{GCH} \iff \neg \text{CTP}(\lambda^+)$. 

Equivalently
For every uncountable cardinal $\lambda$, if $\text{GCH} + \Box_{\lambda^+}(\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ preserves cardinals.

Theorem (Jensen, 1972)
\[ \Box_{\lambda}(\lambda^+, < \lambda^+) \text{ holds iff there exists a special } \lambda^+\text{-Aronszajn tree.} \]

Theorem (Todorcevic, 1987)
\[ \Box_{\kappa}(\kappa, < \kappa) \text{ holds iff there exists a } \kappa\text{-Aronszajn tree.} \]
Square principles

Recall our conjecture
For every uncountable cardinal $\lambda$, $\text{GCH} \iff \neg \text{CTP}(\lambda^+)$.  

Equivalently
For every uncountable cardinal $\lambda$, if $\text{GCH} + \square_{\lambda^+} (\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ preserves cardinals.

Theorem (Ben-David and Shelah, 1986)
For every singular cardinal $\lambda$, if $\text{GCH} + \square_{\lambda} (\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ is $\lambda$-distributive.
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_{\xi}(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.
- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_{\xi}(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_{\xi}(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.
- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_{\xi}(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Jensen exploits the fact that $\Box_{\lambda}(\lambda^+)$ yields a non-reflecting stationary set $S$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.  
- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.  

The constructions under $\xi = \lambda$ use this assumption crucially:

- Jensen exploits the fact that $\Box_\lambda(\lambda^+)$ yields a non-reflecting stationary set $S$. The definition of limit level $T_\alpha$ for $\alpha \in S$ involves throwing away many canonical limits from $\bigcup_{\beta < \alpha} T_\beta$. 

To sum up

A problem of a similar flavor

- Jensen constructed a \( \lambda^+ \)-Souslin tree from GCH + \( \square_\xi(\lambda^+) \) with \( \xi = \lambda \), and we relaxed it to \( \xi = \lambda^+ \).

- Ben-David and Shelah constructed a non-collapsing \( \lambda^+ \)-Aronszajn tree from GCH + \( \square_\xi(\lambda^+, < \lambda^+) \) with \( \xi = \lambda \), and we want to relax it to \( \xi = \lambda^+ \).

The constructions under \( \xi = \lambda \) use this assumption crucially:

- Jensen exploits the fact that \( \square_\lambda(\lambda^+) \) yields a non-reflecting stationary set \( S \). The definition of limit level \( T_\alpha \) for \( \alpha \in S \) involves throwing away many canonical limits from \( \bigcup_{\beta < \alpha} T_\beta \).

By \( \diamond(S) \), this ensures the sealing of antichains.
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_{\xi}(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_{\xi}(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Jensen exploits the fact that $\Box_{\lambda}(\lambda^+)$ yields a non-reflecting stationary set $S$. The definition of limit level $T_\alpha$ for $\alpha \in S$ involves throwing away many canonical limits from $\bigcup_{\beta < \alpha} T_\beta$. By $\lozenge(S)$, this ensures the sealing of antichains.

This does not jam the later stages of the construction, since (one can arrange that) $\text{acc}(C_\alpha) \cap S = \emptyset$ for all $\alpha$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.
- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Ben-David and Shelah exploits the fact that for $\lambda$ singular, $\Box_\lambda(\lambda^+, < \lambda^+)$ may be witnessed by a sequence $\langle C_\alpha \mid \alpha < \lambda^+ \rangle$ for which $\{\alpha < \lambda^+ \mid |C_\alpha| = |\alpha|\}$ is nonstationary.
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \square_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \square_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Ben-David and Shelah exploits the fact that for $\lambda$ singular, $\square_\lambda(\lambda^+, < \lambda^+)$ may be witnessed by a sequence $\langle C_\alpha \mid \alpha < \lambda^+ \rangle$ for which $|C_\alpha| < \lambda$ for all $\alpha < \lambda^+$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \square_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \square_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Ben-David and Shelah exploits the fact that for $\lambda$ singular, $\square_\lambda(\lambda^+, < \lambda^+)$ may be witnessed by a sequence $\langle C_\alpha \mid \alpha < \lambda^+ \rangle$ for which $|C_\alpha| < \lambda$ for all $\alpha < \lambda^+$.

The definition of limit level $T_\alpha$ involves throwing away one canonical limit from $\bigcup_{\beta < \alpha} T_\beta$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi (\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi (\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially:

- Ben-David and Shelah exploits the fact that for $\lambda$ singular, $\Box_\lambda (\lambda^+, < \lambda^+)$ may be witnessed by a sequence $\langle C_\alpha \mid \alpha < \lambda^+ \rangle$ for which $|C_\alpha| < \lambda$ for all $\alpha < \lambda^+$.

The definition of limit level $T_\alpha$ involves throwing away one canonical limit from $\bigcup_{\beta < \alpha} T_\beta$. By $\Diamond (\lambda^+)$, this ensures the sealing of a cofinal branch.
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$. 

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$. 

The constructions under $\xi = \lambda$ use this assumption crucially:

- Ben-David and Shelah exploits the fact that for $\lambda$ singular, $\Box_\lambda(\lambda^+, < \lambda^+)$ may be witnessed by a sequence $\langle C_\alpha \mid \alpha < \lambda^+ \rangle$ for which $|C_\alpha| < \lambda$ for all $\alpha < \lambda^+$. 

The definition of limit level $T_\alpha$ involves throwing away one canonical limit from $\bigcup_{\beta < \alpha} T_\beta$. This does not jam the later stages of the construction, since they build a $\lambda$-splitting tree, while $|C_\alpha| < \lambda$ for all $\alpha$. 
To sum up

A problem of a similar flavor

- Jensen constructed a $\lambda^+$-Souslin tree from $\text{GCH} + \Box_\xi(\lambda^+)$ with $\xi = \lambda$, and we relaxed it to $\xi = \lambda^+$.

- Ben-David and Shelah constructed a non-collapsing $\lambda^+$-Aronszajn tree from $\text{GCH} + \Box_\xi(\lambda^+, < \lambda^+)$ with $\xi = \lambda$, and we want to relax it to $\xi = \lambda^+$.

The constructions under $\xi = \lambda$ use this assumption crucially.

So, “relaxing $\xi = \lambda$ to $\xi = \lambda^+$”, in fact, amounts to finding a different construction.
Same same, but different
Exercise

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\square_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:
Coherent Souslin trees

Exercise

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\square_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:

- For every cofinal $A \subseteq \kappa$, there is a limit $\alpha < \kappa$ such that $\sup(nacc(C_\alpha \cap A)) = \alpha$. 
Coherent Souslin trees

Exercise

Suppose that $◊(\kappa)$ holds, and there exists a $\square_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:

- For every cofinal $A \subseteq \kappa$, there is a limit $\alpha < \kappa$ such that $\sup(\text{nacc}(C_\alpha) \cap A) = \alpha$.

Then there exists a $\kappa$-Souslin tree.
Coherent Souslin trees

Exercise

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\Box_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:

- For every cofinal $A \subseteq \kappa$, there is a limit $\alpha < \kappa$ such that $\sup(\text{nacc}(C_\alpha) \cap A) = \alpha$.

Then there exists a $\kappa$-Souslin tree.

For a quick proof

See “How to construct a Souslin tree the right way” on my webpage.
Coherent Souslin trees

Proposition (Brodsky-Rinot, 2017)

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\square_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:

- For every sequence $\langle A_i \mid i < \kappa \rangle$ of cofinal subsets of $\kappa$, there is a limit $\alpha < \kappa$ such that $\sup(nacc(C_\alpha) \cap A_i) = \alpha$ for all $i < \alpha$. 
Coherent Souslin trees

Proposition (Brodsky-Rinot, 2017)

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\square_\kappa(\kappa)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:

▶ For every sequence $\langle A_i \mid i < \kappa \rangle$ of cofinal subsets of $\kappa$, there is a limit $\alpha < \kappa$ such that $\sup(\text{nacc}(C_\alpha) \cap A_i) = \alpha$ for all $i < \alpha$.

Then there exists a coherent $\kappa$-Souslin tree.
Coherent Souslin trees

Proposition (Brodsky-Rinot, 2017)

Suppose that ♠(κ) holds, and there exists a □(κ)-sequence ⟨C_α | α < κ⟩ satisfying the following:

► For every sequence ⟨A_i | i < κ⟩ of cofinal subsets of κ, there is a limit α < κ such that sup(nacc(C_α) ∩ A_i) = α for all i < α.

Then there exists a coherent κ-Souslin tree.

Note

Wlog, the A_i's are pairwise disjoint. Therefore, |C_α| = |α|.
Proposition (Brodsky-Rinot, 2017)

Suppose that ♦(κ) holds, and there exists a □ₖ(κ)-sequence ⟨C_α | α < κ⟩ satisfying the following:

- For every sequence ⟨A_i | i < κ⟩ of cofinal subsets of κ, there is a limit α < κ such that sup(nacc(C_α) ∩ A_i) = α for all i < α.

Then there exists a coherent κ-Souslin tree.

About the proof

Uses the microscopic approach for Souslin-tree constructions.
Proposition (Brodsky-Rinot, 201∞)

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\Box_\kappa(\kappa, < \kappa)$-sequence $\vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle$ satisfying the following:
Proposition (Brodsky-Rinot, 201∞)

Suppose that ♦(κ) holds, and there exists a □κ(κ, < κ)-sequence \( \vec{C} = \langle C_\alpha | \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \text{sup}(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).
Proposition (Brodsky-Rinot, 201∞)

Suppose that ♦(κ) holds, and there exists a □κ(κ, < κ)-sequence \( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Recall

\[ C_\alpha := \{ C_\beta \cap \alpha \mid \beta < \kappa, \sup(C_\beta \cap \alpha) = \alpha \}. \]
Distributive Aronszajn trees

Proposition (Brodsky-Rinot, 201∞)

Suppose that $\diamondsuit(\kappa)$ holds, and there exists a $\Box_\kappa(\kappa, < \kappa)$-sequence $\vec{C} = \langle C_\alpha | \alpha < \kappa \rangle$ satisfying the following:

- For every club $E \subseteq \kappa$, there is a limit $\alpha < \kappa$ such that $\sup(nacc(C) \cap E) = \alpha$ for all $C \in C_\alpha$.

Then there exists a corresponding tree $T(\vec{C})$ which is $\kappa$-Aronszajn.
Proposition (Brodsky-Rinot, 201∞)

Suppose that ♦(κ) holds, and there exists a □κ(κ<κ)-sequence
\( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Then there exists a corresponding tree \( T(\vec{C}) \) which is \( \kappa \)-Aronszajn.

Note

Ben-David and Shelah used ♦(κ) to seal cofinal branches.
We use club-guessing, instead.
(Instead of throwing away canonical limits, we inject noise)
Proposition (Brodsky-Rinot, 2020)

Suppose that \( \Diamond(\kappa) \) holds, and there exists a \( \square_\kappa(\kappa, < \kappa) \)-sequence \( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Then there exists a corresponding tree \( T(\vec{C}) \) which is \( \kappa \)-Aronszajn. Furthermore, for every cardinal \( \theta \), if the following holds:

- For every sequence \( \langle A_i \mid i < \theta \rangle \) of cofinal subsets of \( \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C_\alpha) \cap A_i) = \alpha \) for all \( i < \theta \).
Distributive Aronszajn trees

Proposition (Brodsky-Rinot, 201∞)
Suppose that ♦(κ) holds, and there exists a □κ(κ, < κ)-sequence
\( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Then there exists a corresponding tree \( T(\vec{C}) \) which is \( \kappa \)-Aronszajn. Furthermore, for every cardinal \( \theta \), if the following holds:

- For every sequence \( \langle A_i \mid i < \theta \rangle \) of cofinal subsets of \( \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C_\alpha) \cap A_i) = \alpha \) for all \( i < \theta \).

Then \( T(\vec{C}) \) is \( \theta \)-distributed.
Distributive Aronszajn trees

**Proposition (Brodsky-Rinot, 201∞)**

Suppose that ♦(κ) holds, and there exists a □κ<κ)-sequence
\[ \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \]
satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(nacc(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Then there exists a corresponding tree \( T(\vec{C}) \) which is \( \kappa \)-Aronszajn.

Furthermore, for every cardinal \( \theta \), if the following holds:

- For every sequence \( \langle A_i \mid i < \theta \rangle \) of cofinal subsets of \( \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(nacc(C_\alpha) \cap A_i) = \alpha \) for all \( i < \theta \).

Then \( T(\vec{C}) \) is \( \theta \)-distributive.
Distributive Aronszajn trees

Proposition (Brodsky-Rinot, 201∞)
Suppose that ◊(κ) holds, and there exists a □κ(κ, < κ)-sequence
\( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).

Then there exists a corresponding tree \( T(\vec{C}) \) which is $\kappa$-Aronszajn. Furthermore, for every cardinal \( \theta \), if the following holds:

- For every sequence \( \langle A_i \mid i < \theta \rangle \) of cofinal subsets of \( \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C_\alpha) \cap A_i) = \alpha \) for all \( i < \theta \).

Then \( T(\vec{C}) \) is $\theta$-distributive.

About the proof
Uses walks on ordinals.
Proposition (Brodsky-Rinot, 201∞)  
Suppose that ♦(κ) holds, and there exists a □κ(κ, < κ)-sequence  
\( \vec{C} = \langle C_\alpha \mid \alpha < \kappa \rangle \) satisfying the following:  

- For every club \( E \subseteq \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C) \cap E) = \alpha \) for all \( C \in C_\alpha \).  

Then there exists a corresponding tree \( T(\vec{C}) \) which is \( \kappa \)-Aronszajn. Furthermore, for every cardinal \( \theta \), if the following holds:  

- For every sequence \( \langle A_i \mid i < \theta \rangle \) of cofinal subsets of \( \kappa \), there is a limit \( \alpha < \kappa \) such that \( \sup(\text{nacc}(C_\alpha) \cap A_i) = \alpha \) for all \( i < \theta \).  

Then \( T(\vec{C}) \) is \( \theta \)-distributive.  

About the proof  
Uses walks on ordinals.  
From \( \vec{C} \), we cook up \( \vec{D} \), and then the tree \( T(\vec{C}) \) is \( T(\rho_0 \vec{D}) \).
To sum up

There are a few machines that take $\square_\xi(\kappa, < \mu)$-sequences $\vec{C}$ as inputs, and produce corresponding trees $T(\vec{C})$ as outputs. We already mentioned two:

- The microscopic approach for Souslin-tree constructions;
- Walks on ordinals.
To sum up

There are a few machines that take $\Box_\xi(\kappa, < \mu)$-sequences $\vec{C}$ as inputs, and produce corresponding trees $T(\vec{C})$ as outputs. We already mentioned two:

- The microscopic approach for Souslin-tree constructions;
- Walks on ordinals.

Whether the outcome tree $T(\vec{C})$ is Aronszajn/Souslin/Collapsing... depends on further features of $\vec{C}$. 
To sum up

There are a few machines that take $\square_\xi(\kappa, < \mu)$-sequences $\vec{C}$ as inputs, and produce corresponding trees $T(\vec{C})$ as outputs. We already mentioned two:

- The microscopic approach for Souslin-tree constructions;
- Walks on ordinals.

Whether the outcome tree $T(\vec{C})$ is Aronszajn/Souslin/Collapsing... depends on further features of $\vec{C}$.

So, if we were to use these machines, then we have to find a way to improve the $\vec{C}$'s.
Improve your square
Postprocessing functions

So, someone provides us with a raw $\Box_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?
Postprocessing functions

So, someone provides us with a raw $\square_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

Definition

$\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ is a postprocessing function iff for all $x \in \mathcal{K}(\kappa)$:
Postprocessing functions

So, someone provides us with a raw $\Box_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

Definition

$\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ is a postprocessing function iff for all $x \in \mathcal{K}(\kappa)$:

Recall

$x \in \mathcal{K}(\kappa)$ iff $x$ is a club in some limit ordinal $\alpha \leq \kappa$. 
Postprocessing functions

So, someone provides us with a raw $\square_{\xi}(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

**Definition**

$\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ is a **postprocessing function** iff for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;

**Recall**

$x \in \mathcal{K}(\kappa)$ iff $x$ is a club in some limit ordinal $\alpha \leq \kappa$. 
Postprocessing functions

So, someone provides us with a raw $\square_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

**Definition**

$\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ is a postprocessing function iff for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;

**Recall**

$x \in \mathcal{K}(\kappa)$ iff $x$ is a club in some limit ordinal $\alpha \leq \kappa$. 
Postprocessing functions

So, someone provides us with a raw $\square_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

**Definition**

$\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ is a **postprocessing function** iff for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\sup(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

**Recall**

$x \in \mathcal{K}(\kappa)$ iff $x$ is a club in some limit ordinal $\alpha \leq \kappa$. 
Postprocessing functions

So, someone provides us with a raw $\square^\xi_\kappa(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

**Definition**

$\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ is a **postprocessing function** iff for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

By convention, let $\Phi(x) := \{\text{sup}(x)\}$ for all $x \in \mathcal{P}(\kappa) \setminus \mathcal{K}(\kappa)$. 
Postprocessing functions

So, someone provides us with a raw $\square_\xi(\kappa, < \mu)$-sequence $\langle C_\alpha \mid \alpha < \kappa \rangle$. How do we proceed?

**Definition**

$\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ is a **postprocessing function** iff for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

By convention, let $\Phi(x) := \{\text{sup}(x)\}$ for all $x \in \mathcal{P}(\kappa) \setminus \mathcal{K}(\kappa)$.

**Lemma (Brodsky-Rinot, 201∞)**

If $\vec{\mathcal{C}} = \langle C_\alpha \mid \alpha < \kappa \rangle$ is a $\square_\xi(\kappa, < \mu)$-sequence, and $\min\{\xi, \mu\} < \kappa$, then $\vec{\mathcal{C}}^\Phi := \langle \Phi(C_\alpha) \mid \alpha < \kappa \rangle$ is a $\square_\xi(\kappa, < \mu)$-sequence, as well.
Postprocessing functions (cont.)

Definition

\( \Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa) \) is a postprocessing function iff for all \( x \in \mathcal{K}(\kappa) \):

- \( \Phi(x) \) is a club in \( \text{sup}(x) \);
- \( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \);
- for every \( \bar{\alpha} \in \text{acc}(\Phi(x)) \), \( \Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha}) \).

So the collection of postprocessing functions forms a monoid that acts on the class of square sequences.
Postprocessing functions (cont.)

**Definition**
\[ \Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa) \] is a **postprocessing function** iff for all \( x \in \mathcal{K}(\kappa) \):
- \( \Phi(x) \) is a club in \( \text{sup}(x) \);
- \( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \);
- for every \( \bar{\alpha} \in \text{acc}(\Phi(x)) \), \( \Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha}) \).

So the collection of postprocessing functions forms a monoid that acts on the class of square sequences. This means that we can start with an arbitrary square sequence \( \vec{C} \); then move to \( \vec{C}^{\Phi_0} \), and then to \( \vec{C}^{\Phi_1 \circ \Phi_0} \), and hopefully, after finitely many steps, we will end up with a useful sequence \( \vec{C}^{\Phi_n \circ \cdots \circ \Phi_0} \).
Postprocessing functions (cont.)

Definition
Φ : \( \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa) \) is a \textbf{postprocessing function} iff for all \( x \in \mathcal{K}(\kappa) \):

- \( \Phi(x) \) is a club in \( \text{sup}(x) \);
- \( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \);
- for every \( \bar{\alpha} \in \text{acc}(\Phi(x)) \), \( \Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha}) \).

So the collection of postprocessing functions forms a monoid that acts on the class of square sequences.

This means that we can start with an arbitrary square sequence \( \vec{C} \); then move to \( \vec{C}^{\Phi_0} \), and then to \( \vec{C}^{\Phi_1 \circ \Phi_0} \), and hopefully, after finitely many steps, we will end up with a useful sequence \( \vec{C}^{\Phi_n \circ \cdots \circ \Phi_0} \).

Our current practical record stands on \( n = 11 \).
Definition

\( \Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa) \) is a **postprocessing function** iff for all \( x \in \mathcal{K}(\kappa) \):

- \( \Phi(x) \) is a club in \( \text{sup}(x) \);
- \( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \);
- for every \( \bar{\alpha} \in \text{acc}(\Phi(x)) \), \( \Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha}) \).

So the collection of postprocessing functions forms a monoid that acts on the class of square sequences.

This means that we can start with an arbitrary square sequence \( \vec{C} \); then move to \( \vec{C}^{\Phi_0} \), and then to \( \vec{C}^{\Phi_1 \circ \Phi_0} \), and hopefully, after finitely many steps, we will end up with a useful sequence \( \vec{C}^{\Phi_n \circ \cdots \circ \Phi_0} \).

Our current practical record stands on \( n = 11 \).

**Question**

What kind of postprocessing functions are there?
List of postprocessing functions
Postprocessing functions - example #1

Recall (postprocessing function)

A map $\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ satisfying for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

For all $x \in \mathcal{K}(\kappa)$, let:

$$\Phi(x) := \text{acc}(x).$$
Postprocessing functions - example #1

Recall (postprocessing function)
A map $\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ satisfying for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

For all $x \in \mathcal{K}(\kappa)$, let:

$$\Phi(x) := \text{acc}(x).$$

Well, the preceding doesn't quite work. Here is how it's done:

$$\Phi(x) := \begin{cases} 
\text{acc}(x), & \text{if } \text{sup}([\text{acc}(x)]) = \text{sup}(x); \\
\end{cases}$$
Postprocessing functions - example #1

Recall (postprocessing function)
A map $\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ satisfying for all $x \in \mathcal{K}(\kappa)$:

- $\Phi(x)$ is a club in $\text{sup}(x)$;
- $\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$;
- for every $\bar{\alpha} \in \text{acc}(\Phi(x))$, $\Phi(x) \cap \bar{\alpha} = \Phi(x \cap \bar{\alpha})$.

For all $x \in \mathcal{K}(\kappa)$, let:

$$\Phi(x) := \text{acc}(x).$$

Well, the preceding doesn't quite work. Here is how it's done:

$$\Phi(x) := \begin{cases} 
\text{acc}(x), & \text{if } \text{sup}(\text{acc}(x)) = \text{sup}(x); \\
\text{x \setminus sup(acc(x))}, & \text{otherwise}.
\end{cases}$$
For some fixed $\epsilon < \kappa$:

$$
\Phi(x) := \begin{cases} 
\{ \alpha \in x \mid \text{otp}(x \cap \alpha) > \epsilon \}, & \text{if otp}(x) > \epsilon; \\
x, & \text{otherwise}.
\end{cases}
$$
Postprocessing functions - example #2

For some fixed $\epsilon < \kappa$:

$$\Phi(x) := \begin{cases} \{ \alpha \in x \mid \text{otp}(x \cap \alpha) > \epsilon \}, & \text{if otp}(x) > \epsilon; \\ x, & \text{otherwise.} \end{cases}$$

More generally, for a fixed closed subset $\Sigma$ of $\kappa$:

$$\Phi(x) := \begin{cases} \{ \alpha \in x \mid \text{otp}(x \cap \alpha) \in \Sigma \}, & \text{if otp}(x) = \sup(\Sigma \cap \text{otp}(x)); \\ x \setminus (x(\sup(\Sigma \cap \text{otp}(x))))), & \text{otherwise.} \end{cases}$$
Postprocessing functions - example #2

For some fixed $\epsilon < \kappa$:

$$\Phi(x) := \begin{cases} \{ \alpha \in x \mid \otp(x \cap \alpha) > \epsilon \}, & \text{if } \otp(x) > \epsilon; \\ x, & \text{otherwise}. \end{cases}$$

More generally, for a fixed closed subset $\Sigma$ of $\kappa$:

$$\Phi(x) := \begin{cases} \{ \alpha \in x \mid \otp(x \cap \alpha) \in \Sigma \}, & \text{if } \otp(x) = \sup(\Sigma \cap \otp(x)); \\ x \setminus (x(\sup(\Sigma \cap \otp(x))))), & \text{otherwise}. \end{cases}$$

Applications

A clever choice of $\Sigma$ could transform a $\square^{\xi}(\kappa, < \mu)$-sequence into a $\square^{\xi'}(\kappa, < \mu')$-sequence with $\xi' < \xi$ or $\mu' < \mu$. 
Postprocessing functions - example #3

For some fixed club $D \subseteq \kappa$:

$$\Phi(x) := \begin{cases} D \cap x, & \text{if } \sup(D \cap x) = \sup(x); \\ x \setminus \sup(D \cap x), & \text{otherwise.} \end{cases}$$
Postprocessing functions - example #3

For some fixed club $D \subseteq \kappa$:

$$\Phi(x) := \begin{cases} 
D \cap x, & \text{if } \sup(D \cap x) = \sup(x); \\
 x \setminus \sup(D \cap x), & \text{otherwise.}
\end{cases}$$

Another useful option:

$$\Phi(x) := \begin{cases} 
\{\sup(D \cap \alpha) \mid \alpha \in x\}, & \text{if } \sup(D \cap \sup(x)) = \sup(x); \\
 x \setminus \sup(D \cap \sup(x)), & \text{otherwise}
\end{cases}$$
Postprocessing functions - example #3

For some fixed club $D \subseteq \kappa$:

$$\Phi(x) := \begin{cases} D \cap x, & \text{if } \sup(D \cap x) = \sup(x); \\ x \setminus \sup(D \cap x), & \text{otherwise.} \end{cases}$$

Another useful option:

$$\Phi(x) := \begin{cases} \{\sup(D \cap \alpha) \mid \alpha \in x\}, & \text{if } \sup(D \cap \sup(x)) = \sup(x); \\ x \setminus \sup(D \cap \sup(x)), & \text{otherwise.} \end{cases}$$

Applications

A clever choice of $D$ could equip a $\square_\xi(\kappa, < \mu)$-sequence with some club-guessing features.
For some fixed $A \subseteq \kappa$:

$$\Phi(x) := \begin{cases} 
\text{cl}(\text{nacc}(x) \cap A), & \text{if } \sup(\text{nacc}(x) \cap A) = \sup(x); \\
\setminus \sup(\text{nacc}(x) \cap A), & \text{otherwise}.
\end{cases}$$
For some fixed $A \subseteq \kappa$:

$$
\Phi(x) := \begin{cases} 
\text{cl}(\text{nacc}(x) \cap A), & \text{if } \sup(\text{nacc}(x) \cap A) = \sup(x); \\
\text{x \setminus sup}(\text{nacc}(x) \cap A), & \text{otherwise.}
\end{cases}
$$

Applications

A dichotomy argument could provide $A$ that would transform a $\Box_\xi(\kappa, < \mu)$-sequence into a $\Box_{\xi'}(\kappa, < \mu)$-sequence with $\xi' < \xi$. 
Postprocessing functions - example #5

Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^{\lambda^+}_{\not= \text{cf}(\lambda)}$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \to \mathcal{K}(\lambda^+)$ satisfying the following.
Postprocessing functions - example #5

Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^{\lambda^+}_{\not\subseteq \text{cf}(\lambda)}$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \to \mathcal{K}(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $\text{nacc}(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$. 
Postprocessing functions - example #5

Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^{\lambda^+} \neq \text{cf}(\lambda)$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \to \mathcal{K}(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $\text{nacc}(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$.

Corollary (Shelah, 2010)

If $2^\lambda = \lambda^+$, then $\diamond(S)$ holds for every stationary $S \subseteq E^{\lambda^+} \neq \text{cf}(\lambda)$. 
Postprocessing functions - example #5

Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^\lambda_{\neq \text{cf}(\lambda)}$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : K(\lambda^+) \to K(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $\text{nacc}(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$.

Corollary (Shelah, 2010)

If $2^\lambda = \lambda^+$, then $\diamond(S)$ holds for every stationary $S \subseteq E^\lambda_{\neq \text{cf}(\lambda)}$.

Corollary (Zeman, 2010)

For $\lambda$ singular, if $2^\lambda = \lambda^+$ and $\Box^*_\lambda$ holds, then $\diamond(S)$ holds for every $S \subseteq E^\lambda_{\text{cf}(\lambda)}$ that reflects stationarily often.
Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^{\lambda^+} \neq \text{cf}(\lambda)$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \rightarrow \mathcal{K}(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $\text{nacc}(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$.

Not enough for intended applications

Hitting a single cofinal set $A$ is nice, but we need to hit many $A_i$’s.
Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E^{\lambda^+}_{\neq \text{cf}(\lambda)}$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \rightarrow \mathcal{K}(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $nacc(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$.

Lemma (Brodsky-Rinot, 201∞)

Assume $\Diamond(\kappa)$. Then there is a postprocessing $\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ such that every sequence $\langle A_i \mid i < \kappa \rangle$ of cofinal subsets of $\kappa$ may be encoded by a single stationary set $G$. 


Theorem (Brodsky-Rinot, 201∞)

Suppose that $2^\lambda = \lambda^+$, $S \subseteq E_{\not\in \text{cf}(\lambda)}^\lambda$ is stationary, and $\langle C_\alpha \mid \alpha \in S \rangle$ is a sequence such that each $C_\alpha$ is a club in $\alpha$ of order-type $< \alpha$. Then there exists a postprocessing function $\Phi : \mathcal{K}(\lambda^+) \rightarrow \mathcal{K}(\lambda^+)$ satisfying the following.

For every cofinal $A \subseteq \lambda^+$, there exist stationarily many $\alpha \in S$ s.t.:

1. $\text{nacc}(\Phi(C_\alpha)) \subseteq A$;
2. $\text{otp}(\Phi(C_\alpha)) = \text{cf}(\alpha)$.

Lemma (Brodsky-Rinot, 201∞)

Assume $\diamondsuit(\kappa)$. Then there is a postprocessing $\Phi : \mathcal{K}(\kappa) \rightarrow \mathcal{K}(\kappa)$ such that every sequence $\langle A_i \mid i < \kappa \rangle$ of cofinal subsets of $\kappa$ may be encoded by a single stationary set $G$. For all $x \in \mathcal{K}(\kappa)$:

If $\text{nacc}(x) \subseteq G$, then $(\Phi(x))(i + 1) \in A_i$ for all $i < \text{otp}(x)$. 
Corollary (Brodsky-Rinot, 201∞)

Suppose $\langle C_\alpha \mid \alpha < \kappa \rangle$ is a $\Box_\xi(\kappa, < \mu)$-sequence, and $2^{|\xi|} = \kappa$.
For cofinally many $\theta < |\xi|$, there exists a postprocessing function $\Phi_\theta : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ satisfying the following.
For every sequence $\langle A_i \mid i < \theta \rangle$ of cofinal subsets of $\kappa$, there are stat. many $\alpha < \kappa$ s.t. $\sup(\text{nacc}(\Phi_\theta(C_\alpha)) \cap A_i) = \alpha$ for all $i < \theta$.

Lemma (Brodsky-Rinot, 201∞)

Assume $\Diamond(\kappa)$. Then there is a postprocessing $\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ such that every sequence $\langle A_i \mid i < \kappa \rangle$ of cofinal subsets of $\kappa$ may be encoded by a single stationary set $G$. For all $x \in \mathcal{K}(\kappa)$:
If $\text{nacc}(x) \subseteq G$, then $(\Phi(x))(i + 1) \in A_i$ for all $i < \text{otp}(x)$.
Corollary (Brodsky-Rinot, 201∞)
Suppose $\langle C_\alpha \mid \alpha < \kappa \rangle$ is a $\square_\xi(\kappa, < \mu)$-sequence, and $2^{\lvert \xi \rvert} = \kappa$.
For cofinally many $\theta < \lvert \xi \rvert$, there exists a postprocessing function $\Phi_\theta : K(\kappa) \to K(\kappa)$ satisfying the following.
For every sequence $\langle A_i \mid i < \theta \rangle$ of cofinal subsets of $\kappa$, there are stat. many $\alpha < \kappa$ s.t. $\sup(\text{nacc}(\Phi_\theta(C_\alpha)) \cap A_i) = \alpha$ for all $i < \theta$.

Next problem
Each $\theta$ has its own $\Phi_\theta$. We need to integrate them together!
Corollary (Brodsky-Rinot, $201\infty$)

Suppose $\langle C_\alpha \mid \alpha < \kappa \rangle$ is a $\square_{\xi}(\kappa, < \mu)$-sequence, and $2^{|\xi|} = \kappa$.

For cofinally many $\theta < |\xi|$, there exists a postprocessing function $\Phi_\theta : K(\kappa) \to K(\kappa)$ satisfying the following.

For every sequence $\langle A_i \mid i < \theta \rangle$ of cofinal subsets of $\kappa$, there are stat. many $\alpha < \kappa$ s.t. $\sup(nacc(\Phi_\theta(C_\alpha)) \cap A_i) = \alpha$ for all $i < \theta$.

Remark

A statement parallel to the preceding, obtained by replacing $\xi < \kappa$ with $\mu < \kappa$ holds true as well.

(The proof, however, is entirely different)
Mixing postprocessing functions
Mixing postprocessing functions

It turns out that the monoid of postprocessing functions is closed under various mixing operations. We found a few. Here’s one.
Mixing postprocessing functions

It turns out that the monoid of postprocessing functions is closed under various mixing operations. We found a few. Here’s one.

Mixing lemma (Brodsky-Rinot, 201∞)

Suppose \( \langle C_\alpha \mid \alpha < \kappa \rangle \) is a \( \square^\times_\xi(\kappa, < \mu) \)-sequence, \( \min\{\xi, \mu\} < \kappa \).

For every \( \Theta \subseteq \kappa \) and every sequence \( \langle S_\theta \mid \theta \in \Theta \rangle \) of stationary subsets of \( \kappa \), there is a postprocessing function \( \Phi : K(\kappa) \to K(\kappa) \) such that, for cofinally many \( \theta \in \Theta \),
Mixing postprocessing functions

It turns out that the monoid of postprocessing functions is closed under various mixing operations. We found a few. Here’s one.

Mixing lemma (Brodsky-Rinot, 201∞)

Suppose $\langle C_\alpha \mid \alpha < \kappa \rangle$ is a $\square_\xi(\kappa, < \mu)$-sequence, $\min\{\xi, \mu\} < \kappa$. For every $\Theta \subseteq \kappa$ and every sequence $\langle S_\theta \mid \theta \in \Theta \rangle$ of stationary subsets of $\kappa$, there is a postprocessing function $\Phi : \mathcal{K}(\kappa) \to \mathcal{K}(\kappa)$ such that, for cofinally many $\theta \in \Theta$,

$$\hat{S}_\theta := \{ \alpha \in S_\theta \mid \min(\Phi(C_\alpha)) = \theta \}$$

is stationary.
Mixing postprocessing functions

It turns out that the monoid of postprocessing functions is closed under various mixing operations. We found a few. Here’s one.

Mixing lemma (Brodsky-Rinot, 201∞)

Suppose \( \langle C_\alpha | \alpha < \kappa \rangle \) is a \( \square_\xi(\kappa, < \mu) \)-sequence, \( \min\{\xi, \mu\} < \kappa \).

For every \( \Theta \subseteq \kappa \) and every sequence \( \langle S_\theta | \theta \in \Theta \rangle \) of stationary subsets of \( \kappa \), there is a postprocessing function \( \Phi : \kappa(\kappa) \to \kappa(\kappa) \) such that, for cofinally many \( \theta \in \Theta \),

\[
\hat{S}_\theta := \{ \alpha \in S_\theta | \min(\Phi(C_\alpha)) = \theta \}
\]

is stationary.

This means

To each \( \theta \) such that \( \hat{S}_\theta \) is stationary, we may find a corresponding postprocessing function \( \Phi_\theta \), and then we can mix them together letting \( \Phi'(x) = \Phi_\theta(x) \) iff \( \min(\Phi(x)) = \theta \).
An application

Conjecture
For every uncountable cardinal $\lambda$, if $\text{GCH} + \square_{\lambda^+}(\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ preserves cardinals.

Theorem (Brodsky-Rinot, 201∞)
For every singular cardinal $\lambda$, if $\text{GCH} + \square_{\lambda^+}(\lambda^+, < \lambda)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ is $\lambda$-distributive.
Conjecture
For every uncountable cardinal $\lambda$, if GCH $+ \Box_{\lambda^+}(\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ preserves cardinals.

Theorem (Brodsky-Rinot, 201$\infty$)
For every singular cardinal $\lambda$, if GCH $+ \Box_{\lambda^+}(\lambda^+, < \lambda)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ is $\lambda$-distributive.

Corollary
For every uncountable cardinal $\lambda$, if GCH $+ \Box_{\lambda^+}(\lambda^+, < \lambda)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ is $\lambda$-distributive.
An application

Conjecture
For every uncountable cardinal $\lambda$, if $\text{GCH} + \square_{\lambda^+}(\lambda^+, < \lambda^+)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ preserves cardinals.

Theorem (Brodsky-Rinot, 201∞)
For every singular cardinal $\lambda$, if $\text{GCH} + \square_{\lambda^+}(\lambda^+, < \lambda)$ holds, then there is a $\lambda^+$-Aronszajn tree $T$ s.t. $\mathbb{P}(T)$ is $\lambda$-distributive.

An unrelated application of the mixing lemma
If $\square(\kappa)$ holds, then any fat subset of $\kappa$ may be split into $\kappa$ many fat sets.
Blowing up

We have demonstrated the power of postprocessing functions, but there are also some disadvantages.
Blowing up

We have demonstrated the power of postprocessing functions, but there are also some disadvantages.
Most promimently, the requirement “$\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$” prevents us from blowing-up the order-type of elements of a square.

Blowing up

We have demonstrated the power of postprocessing functions, but there are also some disadvantages. Most prominently, the requirement “$\text{acc}(\Phi(x)) \subseteq \text{acc}(x)$” prevents us from blowing-up the order-type of elements of a square. For this, we developed a separate tool. Here is an application.
Blowing up

We have demonstrated the power of postprocessing functions, but there are also some disadvantages. Most prominently, the requirement “\( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \)” prevents us from blowing-up the order-type of elements of a square. For this, we developed a separate tool. Here is an application.

**Theorem (Brodsky-Rinot, 201∞)**

Assume GCH, \( \lambda \) is a singular cardinal, and there is a non-reflecting stationary subset of \( E_{\lambda^+}^{\lambda^+} \neq \text{cf}(\lambda) \).
Blowing up

We have demonstrated the power of postprocessing functions, but there are also some disadvantages. Most prominently, the requirement “\( \text{acc}(\Phi(x)) \subseteq \text{acc}(x) \)” prevents us from blowing-up the order-type of elements of a square. For this, we developed a separate tool. Here is an application.

**Theorem (Brodsky-Rinot, 201∞)**

Assume GCH, \( \lambda \) is a singular cardinal, and there is a non-reflecting stationary subset of \( E^{\lambda^+}_{\neq \text{cf}(\lambda)} \).

If \( \Box^*_\lambda \) holds, then there is a \( \Box_\lambda^2(\lambda^+, < \lambda^+) \)-sequence \( \vec{C} \), for which the microscopic approach to Souslin-tree constructions produces a \( \lambda^+ \)-Souslin tree which is moreover free.